www.ijecs.in
*International Journal Of Engineering And Computer Science ISSN:2319-7242*
*Volume 4 Issue 7 July 2015, Page No. 13096-13100*

# Minimizing Time Complexity based on Parallel Processing using RNS for Audio

***Divya Jennifer Dsouza, Radhakrishna Dodmane***
III$^{nd}$ Final Part-Time MTech, Dept of CSE
NMAMIT, Nitte, Karnataka
India
jenniferdsouza87@nitte.edu.in

Associate Professor, Dept of CSE
NMAMIT, Nitte, Karnataka
India
radhakrishna@nitte.edu.in

***Abstract:*** In this paper, we propose a secure mechanism to transfer sensitive audio files over an unreliable network. In order to prevent the unauthorized access of data that can be collected at the time of data transmission, a secure approach is introduced Audio Cryptography with parallelism that may be some of the future solution for the above mentioned problem. The proposed approach uses the suitable technique such as the Pseudo Random Number Generation(PRNG) method with seed called as the Initial Vector (IV) as the parameter for generating random numbers and the Residual Number System for encryption. The audio file is in turn encrypted using the keys generated using the PRNG. The concept of Residual Number System (RNS) based on Chinese Remainder Theorem (CRT) for share creation is performed in parallel to maximize the resource utilization and make the process faster. These, audio shares are sent over the channel over the network to increase security. Theoretical analyses and experimental results show that, there is significant reduction in time consumption.

*Keywords:* Audio encryption, RNS, Parallel computing, shares, PRNG.

## I. INTRODAUCTION

Sending sensitive documents over the unreliable network is not safe. There are chances that such documents could be easily eavesdropped by any intruder. Hence a secure approach was a need. Cryptography [2] is a technique of either scrambling or transforming the data into an unreadable format called as the cipher text. Only the user who possesses the secret key can decipher it. This paper proposes a novel possible technique named as parallel encryption of Audio using PRNG [13] to produce a multilevel security for the sensitive information transfer.

The first layer of encryption involves generating random keys using the random function using the PRNG which takes Initial Vector as the input to produce encryption key “*KEY*” [7] as output to it. Using this key, a cipher text is generated.

The second layer of encryption involves encrypting the given audio using the “KEY” to produce different shares. These shares are in turn transferred over the network. Using the Chinese Reminder Theorem (CRT), it is possible to encode the audio file. The original audio file is being split up into *n shares*. Decoding could be done by simply stacking up these *n shares*. Audio Cryptographic Technique (ACT) scheme uses (n, n) access threshold structure [1] in which the secret audio is encrypted into n number of shares where n depends on the number of prime numbers used for factorization. Decryption involves combining all shares to reveal the audio at the receiver end.

This paper is divided into the following sections.

Section II explains the literature survey, depicting the existing systems on the audio encryption complexities and the proposed system. Section III describes the problem statement and the complete working of the proposed security system. In Section IV some of the experimental results are shown. Section V gives the conclusions drawn and future work.

## II. LITERATURE SURVEY

With the development of network and multimedia technology, multimedia data such as images, audio-video files are more frequently used in day to day transactions [5]. In order to safeguard the sensitive information on politics, military, research works etc. Many algorithms are introduced and developed. The literature survey elaborates the fact that there is lot of scope for improvement in the field of audio encryption. The most challenging work could be to remove the deficiencies found in the earlier techniques with respect to the consumption of the network bandwidth, unable to decode the message due to some of the shares getting lost during transmission, communication channels getting hacked, failure of the transmission medium, complexity in encoding and decoding of the audio files, latency in transferring the files, unable to utilize the available resources to a fuller extent etc.

The proposed system uses (n, n) scheme of audio encryption, where an audio file is divided into n number of shares using the Chinese remainder algorithm. A novel cryptographic approach combining with parallelism is introduced. As part of the first layer of security, this scheme uses the given input stream for encryption and stores it.

Secondly prime numbers are randomly generated using the Pseudo random technique. These prime numbers are used for the process of encryption. Finally the audio is encrypted to produce cipher text (shares) to protect sensitive data using the RNS algorithm [1] which is based on the Chinese reminder Theorem (CRT). Shares are created using this scheme. The number of shares formed depends on the number of prime numbers being generated. If there are n number of shares created using this scheme, all these shares are stacked together and sent to the destination at the receiver end. It is of low cost and supports direct bit- rate control. The proposed system uses substitution processes, which makes that a change in cipher text cannot spread out to cause great changes. Thus compared with some algorithms consisting of diffusion processes, it is not so sensitive to cipher text changes. Therefore, it is more robust to transmission errors. Also as we use parallelism in our technique combined with PRNG gives us optimal security to the data that needs to be transmitted.

The advantage is the delay in transfer of files could be avoided by encrypting these shares in parallel.

## III. Working of the Proposed System

With the advancement of the multimedia technology, there are enormous amount of information exchange taking place in the network. As this demand is being progressively increasing, there is requirement for strong and efficient yet simple encryption techniques.

Multilevel security is introduced to make the data more secure. The layer uses the Random key [19] to encrypt the sensitive information for its massive combination of keys. The second layer of security comes in when the encryption to the audio medium is being performed using the share creation. This method comprises of encrypting the audio file by dividing it into different shares (say n). The end system can stack up those shares; decrypt them using the RNS's inverse function to obtain the encrypted audio shares. The encrypted audio is further decrypted using the private key to reconstruct the original audio files.

The shares are transferred via different ports to increase security. The disadvantage could be if the hacker could get access to all the channels through which these shares are being transferred, and also the private key to decrypt the encrypted text he/she could gain access to the information. This however is a very complex task.

### A. *Architecture*

As shown in the below Figure 1, the sender first selects the audio file that he wants to transmit securely. Using the Pseudo Random Number Generator the encryption key is being obtained. The PRNG used a random seed called as Initial Vector (IV) [22] as parameter to it. Using this parameter, key is being generated which is used to encrypt using the RNS algorithm. A cipher text is generated in the form of audio shares. The Audio file is divided into *n shares* and transmitted along the channel. The shares are combined at the receiver end. The result is the Encrypted Audio file. The cipher shares are further decrypted using key generated using the PRNG function. The key is used in RNS function applied at the receiver end and the secret audio file is obtained back.

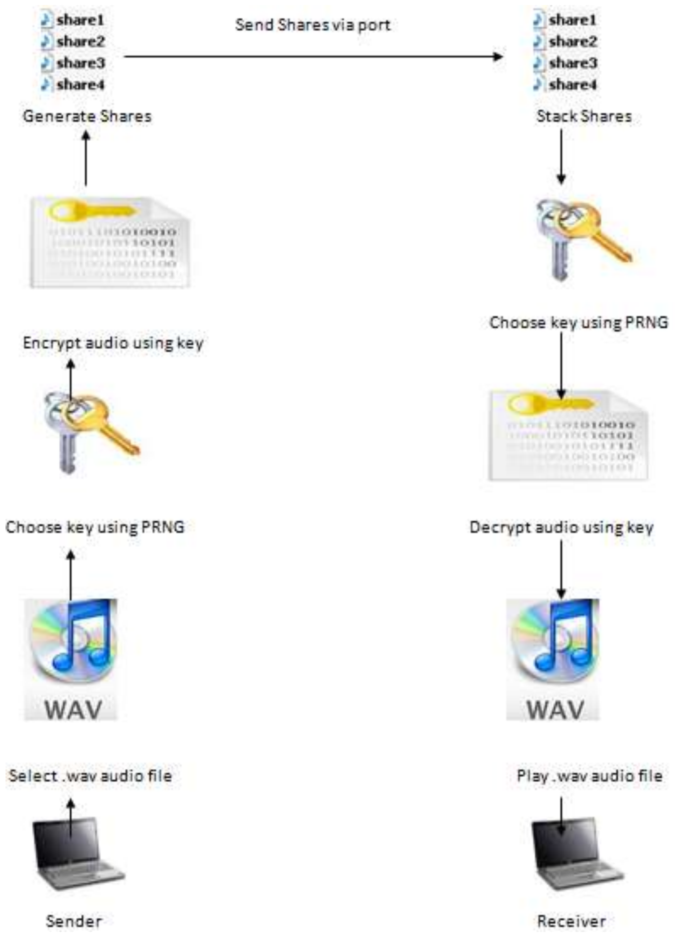


*Figure 1*

### B. *Sender Side Module*

As shown in the below Figure 2, proposed system takes a secret message converts into bytes, and uses them as a stream. The stream obtained is processed in parallel by giving as input to the RNS algorithm. The factors used in the creation of encrypted shares are generated using PRNG to produce a cipher text. The PRNG algorithm uses a nonce [6] to produce random keys. The audio files are processed in streams. The result is the Encrypted Audio file with *n shares*.

The sender processes the shares created. The shares formed are compressed to reduce network bandwidth consumption. The zipped files consisting of the shares are sent over transmission channels. All *n shares* [10] are required to transfer to the receiver.

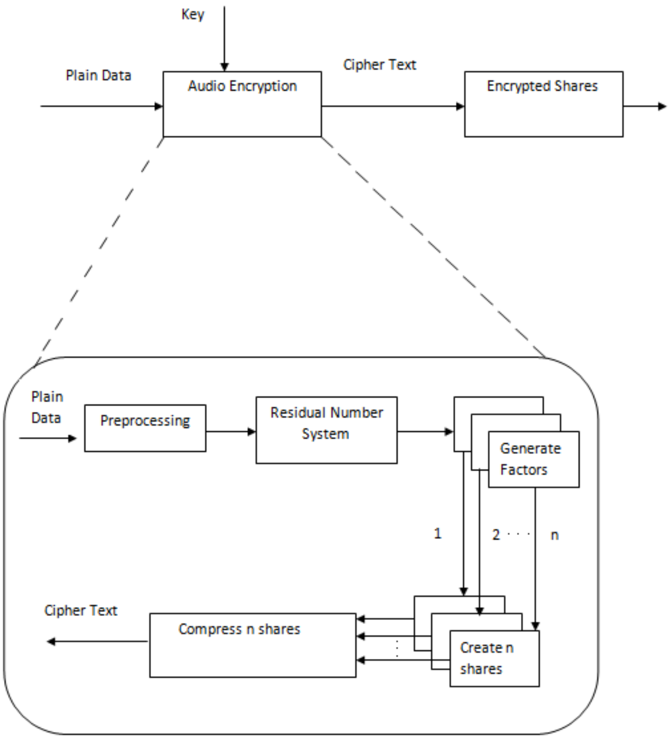


*Figure 2*

### *C. Audio Cryptography Technique(ACT)*

The plain data (audio file) is decomposed into n number of shares such that the original data could be obtained by stacking all *n shares*. Each encrypted shares contains partial encrypted portion of the secret data. Using these shares plain data could be retrieved.

#### I. Steps for creating shares at the Sender Side:

The shares are framed by applying RNS which is based on CRT [7]. This approach uses (n, n) access structure which means *n shares* is required at the receiver end to recreate the original message.

Step 1: Select co-prime numbers ($m_1$, $m_{2,\dots}m_n$).

Step 2: Compute X in the following equation:

$x_i$ =X mod m where xi < $m_i$ and 1< i <n

Step 3: The $X_i$ form the shares. The number of shares depend on how many prime numbers do we take.

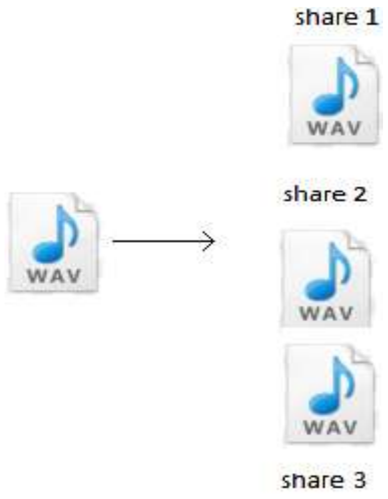


*Figure 3*

#### II Steps for stacking up shares at the Receiver Side:

Step 1: Calculate $\alpha_i = M/m_i$ where $M=m_1.m_2\dots m_n$

Step 2: Find the solution αi.Ti mod $m_i$ Where $T_i$ is multiplicative inverse of $\alpha_i$

Step 3: We can get back original pixel using below eqn:

$$X= (\sum^{n}_{i=1}\boldsymbol{\alpha_i T_i x_i}) \bmod M$$

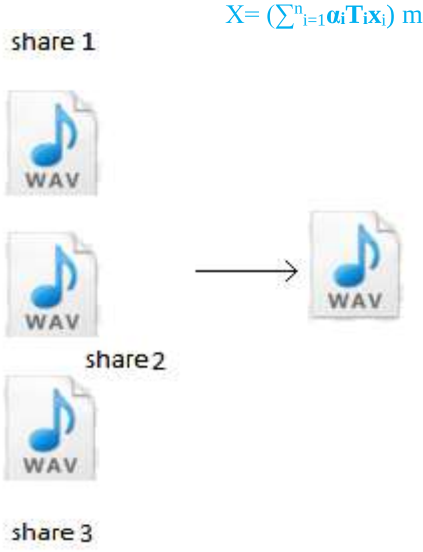


Figure 4

### *D. Receiver End Module*

As shown in the below diagram (Figure 5) recovery of the plain data needs to done. This is done so as to regenerate original message. RNS algorithm is applied with the inverse function using the factors. Appropriate weights are applied to retrieve back the original data.

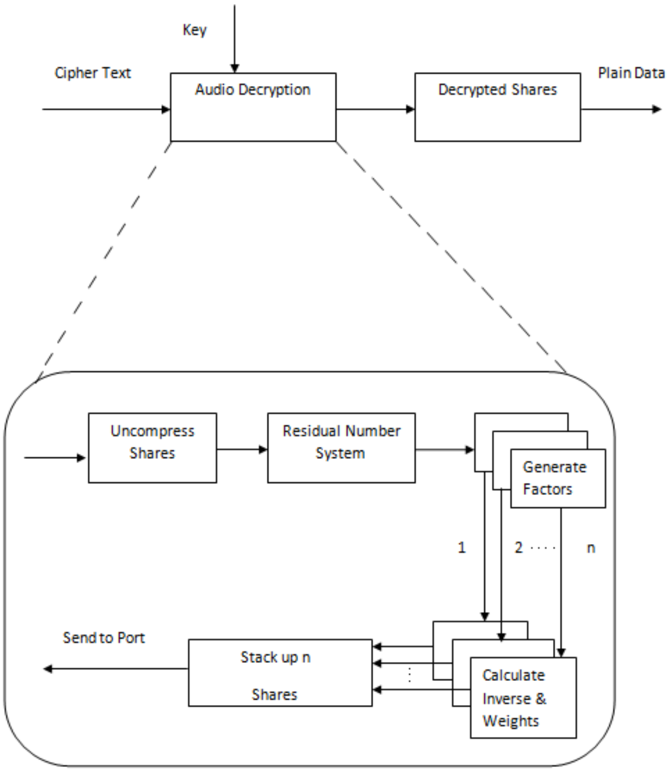


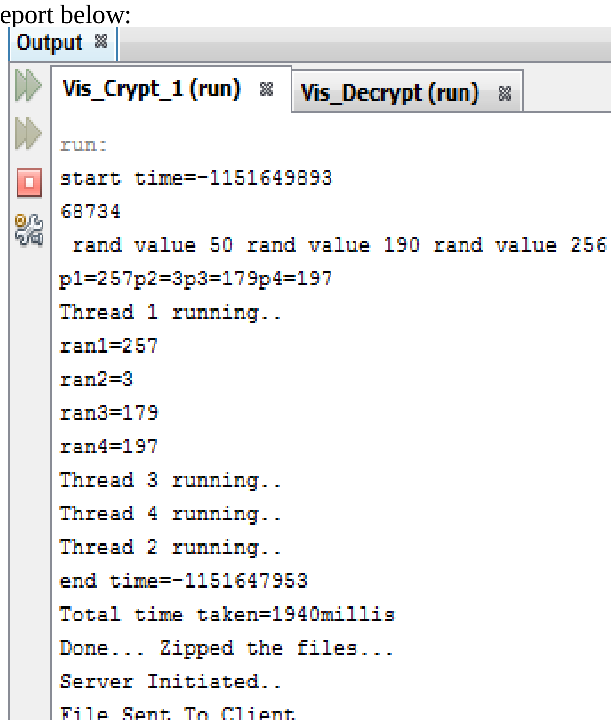
*Figure 5*

## IV. Experimental Results

### *A. At the Server Console:*

Input Audio: Browsing sensitive information (audio file) needed to be transferred.

We encrypt the audio in parallel using the key, generated using PRNG algorithm and calculate the time complexity. The audio is split into four shares in parallel. The total time consumed is 1940 Millis which is shown in the log report below:

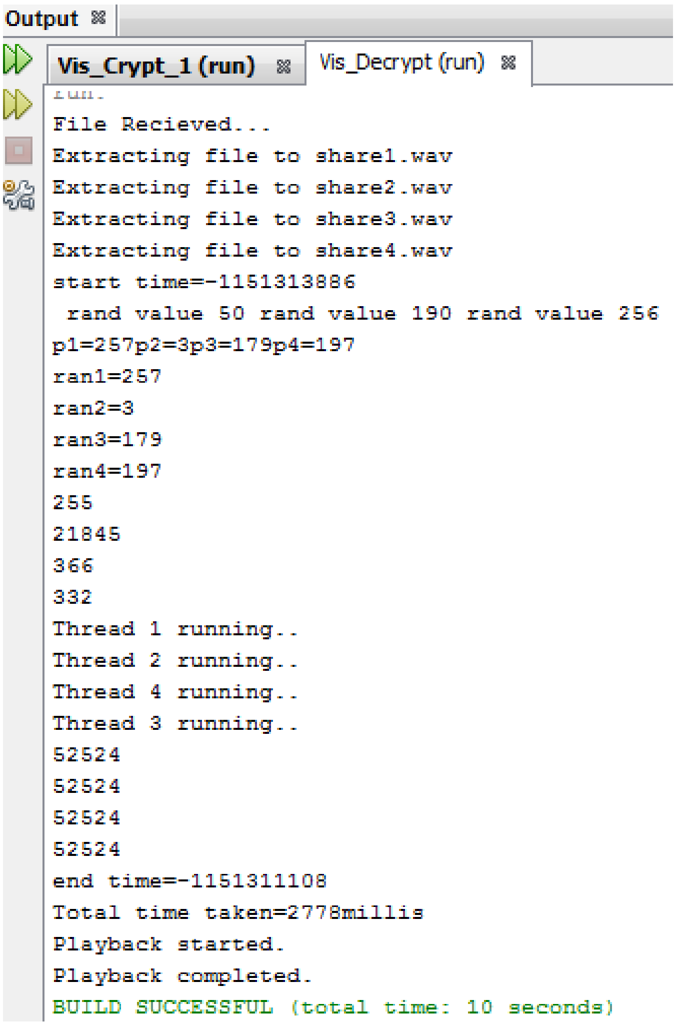


```
run:
start time=-1151649893
68734
 rand value 50 rand value 190 rand value 256
p1=257p2=3p3=179p4=197
Thread 1 running..
ran1=257
ran2=3
ran3=179
ran4=197
Thread 3 running..
Thread 4 running..
Thread 2 running..
end time=-1151647953
Total time taken=1940millis
Done... Zipped the files...
Server Initiated..
File Sent To Client
```

*Figure 6*

### *B. At the Receiver Console*

We can decrypt the shares at the receiver end by processing PRNG [25] and RNS algorithm in the reverse order. Decryptions of shares are also performed in parallel. The received shares are being combined to obtain plain data. This is shown in the log report below:



```
run:
File Recieved...
Extracting file to share1.wav
Extracting file to share2.wav
Extracting file to share3.wav
Extracting file to share4.wav
start time=-1151313886
 rand value 50 rand value 190 rand value 256
p1=257p2=3p3=179p4=197
ran1=257
ran2=3
ran3=179
ran4=197
255
21845
366
332
Thread 1 running..
Thread 2 running..
Thread 4 running..
Thread 3 running..
52524
52524
52524
52524
end time=-1151311108
Total time taken=2778millis
Playback started.
Playback completed.
BUILD SUCCESSFUL (total time: 10 seconds)
```

*Figure 7*

Experimental results show that there is significant reduction in the processing time after applying parallelism to process the shares. The below chart depicts the same:

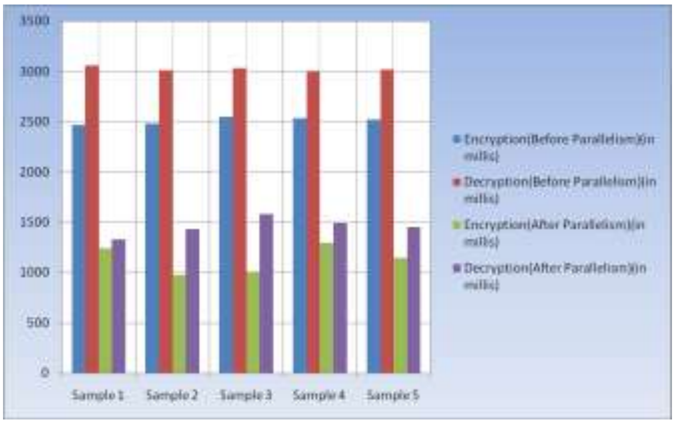


*Figure 8*

### *C. Statistics*

The table shows the mean and standard deviation of the cipher values. As given in the table the SD value is low [11], which the key feature for cryptography.

| Samples | Size(kb) | Mean | Variance | S.D |
|---|---|---|---|---|
| conjested.wav | 52 | 59.77 | 19.73 | 4.44 |
| beep-01a.wav | 62 | 54.94 | 17.03 | 4.13 |
| beep-04.wav | 50 | 55.47 | 17.33 | 4.16 |
| Bike Horn.wav | 68 | 68.77 | 16.7 | 4.09 |
| film.wav | 95 | 57.27 | 13.36 | 3.65 |
| misc012.wav | 359 | 58.73 | 17.4 | 4.17 |
| Pingas.wav | 101 | 58.16 | 17.15 | 4.14 |

*Table 1*

Also, we could find significant difference in the cipher values produced by applying different keys on the same plain data [11]. Even with the change in one bit in the key, there were different cipher values being generated which are a desirable feature. This is shown in the below figure:

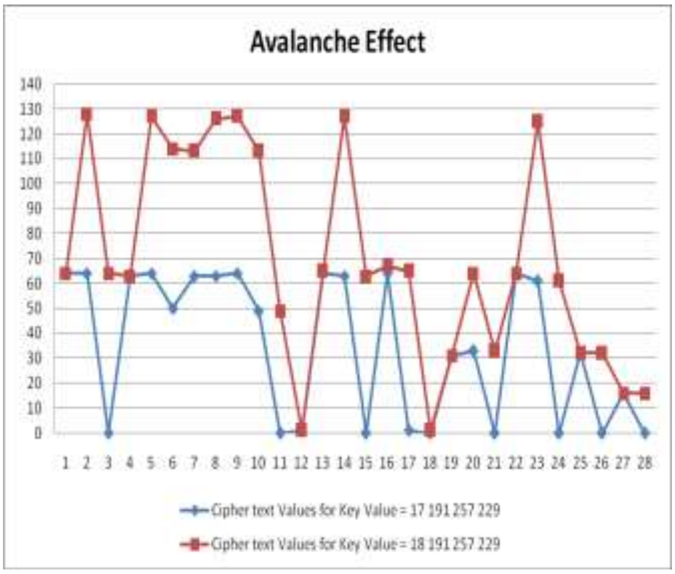


*Figure 9*

## V. CONCLUSIONS

A multilevel approach which includes PRNG and Audio based cryptography technique is being proposed, which is suitable for multimedia encryption and decryption. This method is of low cost, reliable, supports direct bit rate control and is more robust to transmission errors. These characteristics make it more applicable for the current needs. The proposed technique overcomes side channel attacks and algebraic attacks.

## VI. REFERENCES

[1] Andrew Rukhin, J. S. (Revised: August 2008). "A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications" (Revision 1 ed.). Special Publication 800-22.

[2] Ankur, D. V. ( 2014 ). "A New Approach to Pseudorandom Number Generation". Fourth International Conference on Advanced Computing & Communication Technologies.

[3] Bloom, C. A. (1983). "A modular approach to key safeguarding".IEEE transaction on Information Theory, 29(2), pp. 208-210.

[4] Blakley, G. "Safeguarding cryptographic keys " . Proc. Of AFIPS National Computer Conference.**(Article in a conference proceedings)**

[5] Desmedt, Y. (1997). "Some recent research aspects of threshold cryptography". Proc of ISW'97 1st International Information Security Workshop. 1196 of LNCS paper, pp. 158-173 .Springer-Verlag.**(Article in a conference proceedings)**

[6] E. Bohl, M. a. (2014). "A True Random Number Generator with On-Line Testability". 19$^{Th}$ IEEE European Test Symposium (ETS).

[7] (May 2008). GENERATE SEED. In GENERATE SEED macro in php produces 24 bits of entropy and simplifies brute force attacks against the rand and mt rand functions.

[8] ElGuang Zeng, W. H. (n.d.). "Word-Oriented Stream Cipher For 3rd Generation: TAIYI".

[9] Slavco Sajic, B. M. (2013). "RANDOM BINARY SEQUENCES IN TELECOMMUNICATIONS" .Journal of ELECTRICAL ENGINEERING , 64 (4), 230–237. **(Article in a journal)**

[10] SUNEEL, M. ( 2009). "Cryptographic pseudo- random sequences from the chaotic Henon map". (©. I. Sciences., Ed.) 34 ( Part 5), 689–701.**(Article in a Journal)**

[11] Kinga MÁRTON, e. a. (2012). "GENERATION AND TESTING OF RANDOM NUMBERS FOR CRYPTOGRAPHIC APPLICATIONS". In S. A (Ed.), PROCEEDINGS OF THE ROMANIAN ACADEMY,13, Number 4, pp. 368–377. ROMANIAN.

[12] Ranjan Kumar H S, P. K. (2013, July). "Enhanced Security System using Symmetric Encryption and Visual Cryptography". International Journal of Advances in Engineering & Technology,(pp 5-10) **(Article in a Journal).**

[13] Sagar A. Yeshwantrao, P. V. ( 2014). "Shared Cryptographic Scheme with Efficient Data Recovery and compression for Audio Secret Sharing". International Journal of Emerging Technology and Advanced Engineering , 4 (2).**(Article in a Journal).**